DESIGN, IMPLEMENTATION, AND TESTING OF PERTURBATION METHOD FOR HANDWRITTEN NUMERAL RECOGNITION

Thien M. Ha and Horst Bunke
University of Berne
Institut für Informatik und Angewandte Mathematik
Neubrückstr. 10, CH-3012 Bern, Switzerland
Email: {haminh,bunke}@iam.unibe.ch

October 1, 1996

Abstract

This report presents a new approach to off-line handwritten numeral recognition. From the concept of perturbation due to writing habits and instruments, we propose a recognition method which is able to account for a variety of distortions due to eccentric handwriting. The new approach constitutes a shift from the usual pattern recognition paradigm where normalisation is the first step prior to feature extraction and classification. Normalisation is replaced by a set of perturbation processes modelling writing habits and instruments. As a result, the subsequent operations – feature extraction and classification – are independently applied for each perturbation type yielding a set of results that are eventually combined. We tested our method on two worldwide standard databases of isolated numerals, namely, CEDAR and NIST, and obtained 99.09% and 99.54% correct recognition rates at no-rejection level, respectively. The latter result was obtained by testing on more than 170000 numerals.

CR Categories and Subject Descriptors: I.5.0 [Pattern Recognition]: General; I.5.1 [Pattern Recognition]: Models; I.5.2 [Pattern Recognition]: Design Methodology;

Keywords: Optical Character Recognition (OCR), writing habits and styles, writing instruments, reversing process, pattern recognition, perturbation method, decision combination, k-nearest neighbor rule, neural networks.

Contents

1 Introduction

Automatic character recognition is a subfield of pattern recognition and can either be on-line or off-line. On-line recognition refers to those systems where the data to be recognised is input through a tablet digitiser, which acquires in real-time the position of the pen tip as the user writes. In contrast, off-line systems input the data from a document through an acquisition device, such as a scanner or a camera. Off-line character recognition is moreover divided into two categories, namely, machine-printed and handwritten. This report is concerned with one particular problem in off-line handwriting recognition, namely, the recognition of isolated numerals.

Off-line handwritten numeral recognition is an important and necessary step in many document processing applications. For instance, we can think of the reading/processing of checks, mail addresses, tax forms, and census forms. Although the recognition of such documents usually requires more than just that of isolated numerals, the latter remains a necessary and often critical part in most real applications.

In this report we present a new approach to off-line handwritten numeral recognition. Our approach is based on the modelling of human writing habits as well as writing instruments and thus tackles the difficult cases in totally unconstrained data. The writing habits are modelled via a set of geometric transformations (e.g., rotation and slant) whereas the variation due to writing instruments is modelled via the two morphological operations of dilation and erosion. These perturbation models aim at representing the deviations of written numerals from what we consider 'standard patterns'. Assuming that these models cover a large spectrum of writing habits and instruments, their use in the design of a recognition system requires a kind of *reversing process*, capable of correctly bringing a written numeral back to one of its standard forms.

Many approaches have been proposed to solve the reversing problem. For instance, the input image can be normalised according to a set of reversing parameters computed from the image itself [15, 19]. In [27], the reversing parameters are determined so as to equalise a density function. These normalisation procedures have all contributed to improving the recognition rate significantly. However, they include a number of drawbacks that will be detailed in Section 2. Translation- and rotation-invariant transforms have also been proposed as an indirect way to cope with the reversing problem [18, 10]; they are generally limited to simple (geometric affine) transformations and are not complete. Elastic matching is another promising indirect way to cope with this problem. However, it suffers from two shortcomings, namely, that prototypes have to be hand-crafted and that the definition of cost functions is not obvious [17, 25]. The recognition of strongly distorted data has also been tackled by training the recogniser on additional artificially generated data [1, 4, 11, 28]. Further discussions on these different approaches will be presented in Section 6.

In our approach, the reversing process is model-based. That is, in contrast with the usual pattern recognition paradigm, where normalisation is performed purely bottom-up, we first apply a fixed, predefined set of hypothetical inverse perturbations to the image. Then all perturbed versions are submitted separately to a same conventional numeral recogniser. These two steps provide us with a set of results which are subsequently combined. The idea behind this scheme is that at least one of the hypothetical inverse perturbations will correctly bring the input image back to one of its standard forms. Such a perturbation should manifest its evidence by outperforming the others in terms of output score. Thus the reversing process is not exclusively based on image data but utilises model information, too. In this report we limit ourselves to describing the task of recognising the input numeral without identifying the best inverse perturbation model(s) explicitly.

Section 2 presents the basic observations that led us to the approach proposed in this report. The general architecture of our system is described in Section 3. Details of the new recognition method are provided in Section 4. Section 5 presents our experiments using the CEDAR and NIST databases. We discuss our results and comment on relationships of our approach to others in Section 6. Section 7 concludes the report.

2 Observations

The key idea of the perturbation approach proposed in this report lies in the process of reversing an input image back to one of its standard forms. In the present section, we will show, through examples, that

Observation 1
the reversing process can not always be correctly achieved by using the input data alone.

To support our assertion, we will give some examples where a reversing process using input data alone fails. By 'input data alone' we mean all techniques based on the pixel distribution disregarding the nature of the alphabet being considered. For instance, line regression is a powerful method to determine the rotation angle of a character [19]; it is based on pixel distribution and is independent of the alphabet (be it Arabic numerals, Roman characters, Greek characters, etc.). For instance, in Fig. 1a, the detection of the rotation angle by line regression yields a correct result for the numeral '6'. However, in Figs. 1b and 1c, line regression fails to detect the correct rotation angles of the numerals '2' and '4'. Notice that in the case of Fig. 1b, the result would be correct if we were considering the Greek alphabet (symbol γ). These simple observations show that the input data alone may fail to provide the correct reversing parameters. In order to overcome this problem, the reversing process must take into account, among other factors, the nature of the alphabet in use.

Although we have taken the rotation operation as an example, it should be clear that the argumentation is applicable to other geometric transformations, e.g., slant

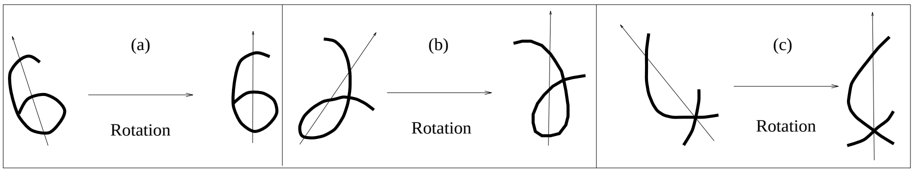

Figure 1: Observations.

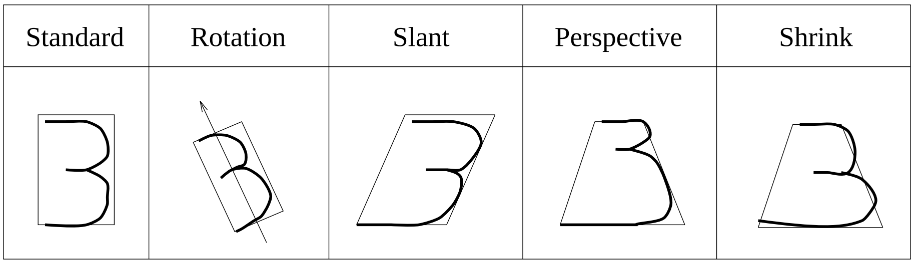

Figure 2: Examples of styles.

and perspective views. In general we observe that

> **Observation 2**
> *most perturbations are due to writing habits (styles) and instruments.*

Without referring in detail to other disciplines, such as graphology and calligraphy, it can readily be seen that a number of simple geometric transformations (such as rotation, slant, perspective view and shrink) can account for a wide variety of handwriting habits and styles (see Fig. 2). In handwriting, the instrument is the pen, whose tip may have various sizes resulting in a variable stroke width. Fig. 3 illustrates the effect of writing the numeral '3' using different pens.

3 Perturbation-based Recognition

In this section we describe the architecture of our recognition scheme, which is based on Observation 1. In order to solve the problem that the reversing process cannot

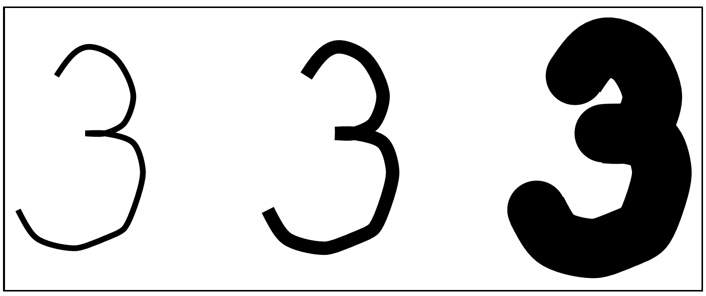

Figure 3: Influence of writing instrument.

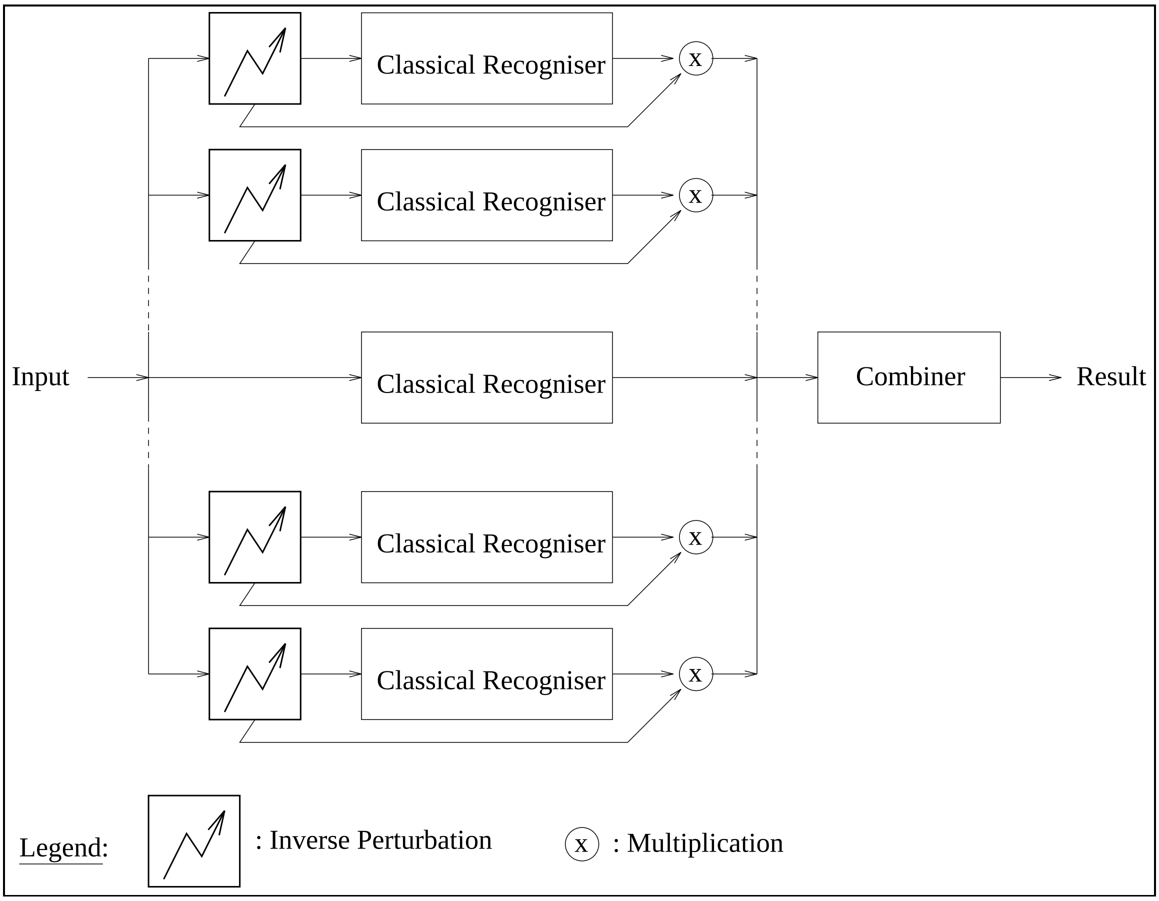

Figure 4: Perturbation-based recognition system.

always be correctly achieved using the input data alone, we propose the reversing process to be model-based, i.e., taking into account the nature of the alphabet being considered. Since we are primarily interested in designing a recognition system and not concerned with explicitly recovering the standard form, the reversing process will be implicit.

The basic idea consists in applying a set of predefined inverse perturbations to the input image (see Fig. 4). These inverse perturbations are independent of the input image and are expected to include the true perturbation that actually made the input image different from its standard pattern. We know that if an inverse perturbation actually corresponds to the true perturbation, the corresponding inverse image will be very close to the original standard pattern and could be easily recognised by some known method. Therefore, each inverse image is submitted separately to a conventional recognition system, the output score of which is then compared to the others. It is clear that among the scores, the one corresponding to the true perturbation can be expected best. Since each score is attached to a class, the recognition scheme is in fact a by-product of the reversing process.

In reality, a written pattern may result from a mixture of many perturbation

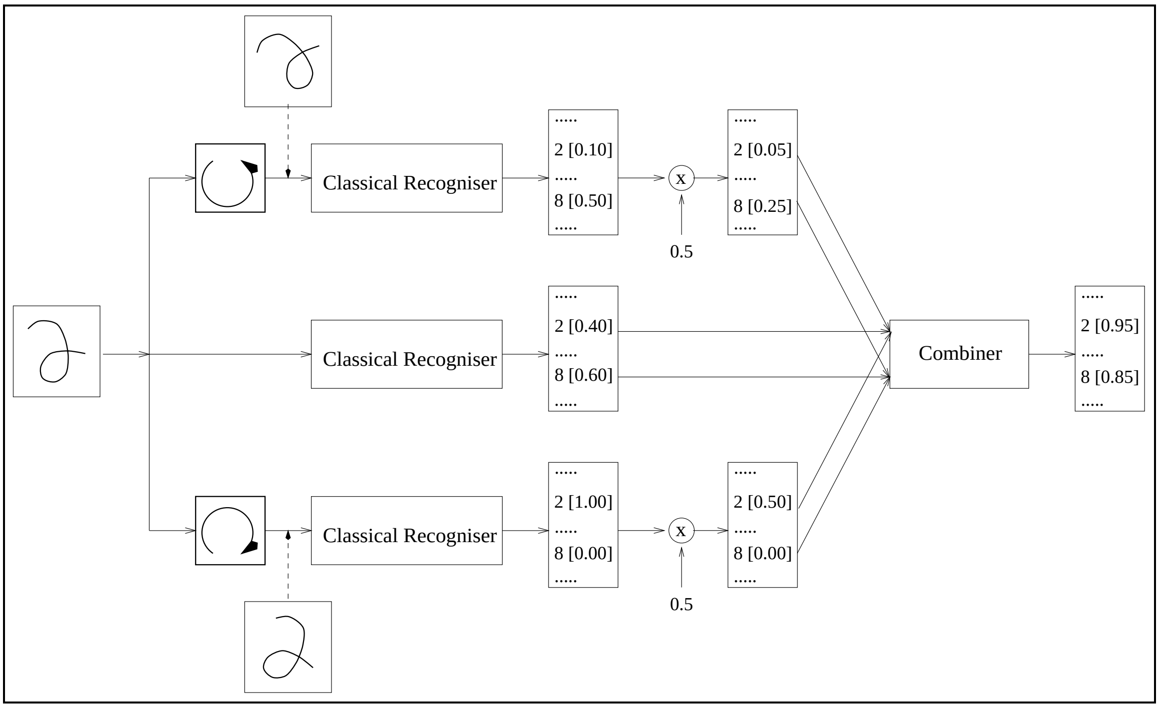

Figure 5: A simple rotation-based perturbation system.

types (mixed style). So the determination of the true perturbation may prove difficult. However, if the goal is to design a recognition system, then a weighted-voting scheme should be sufficient to determine the most probable class of the input image. Notice in Fig. 4 that each output score is down-weighted by a factor proportional to the degree of distortion in order to avoid that an input image is transformed into a completely different pattern without being penalised.

To further illustrate the perturbation method, let us consider a simple example, where only one geometric perturbation type, namely, the rotation is used. The system then espouses the architecture of Fig. 5, where the down-weighting factors are fixed to 0.5. The input image is that of Fig. 1b. It can be seen that the inverse perturbation of the bottom channel is the correct one and thus the output score takes on the value 1.0 for class '2' and 0.0 for class '8'. The output scores of the top channel are more scattered, because the rotation does not bring the input image to anything known by the "classical recogniser". The net result out of the combiner – an adder – is that the bottom channel inverses the likelihood ratio of the "classical recogniser" without perturbation.

4 Parametrisation

The previous section proposed a new recognition scheme in generic form, i.e., without specifying the exact parameters. By using Observation 2 on writing habits, we propose a parametrisation based on four geometric transformations, namely,

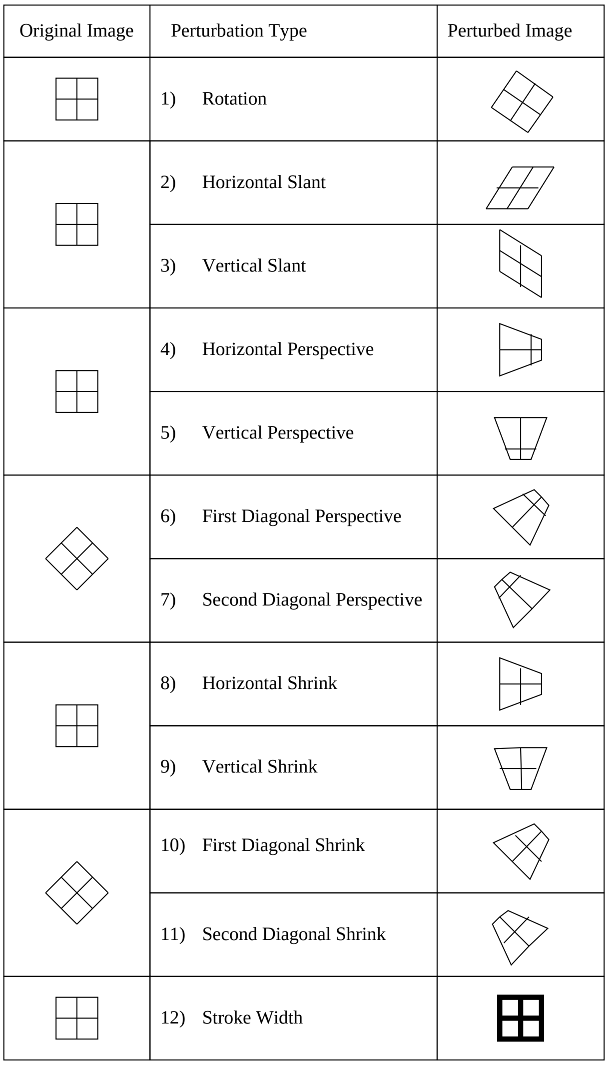

Figure 6: Parametrisation.

rotation, slant, perspective view and shrink. Moreover, slant is decomposed into horizontal and vertical directions, whereas perspective view and shrink are each decomposed into horizontal, vertical, 1^{st} diagonal and 2^{nd} diagonal directions. The perturbations due to writing instruments is modelled by two morphological operators, namely, dilation and erosion. These result in a total of $T = 12$ perturbation types (Fig. 6).

Let $I(x, y)$ be an image and $I'(x', y')$ its perturbed version defined by the perturbation type $t(t = 1, .., T)$ and degree $\delta \in \mathcal{R}$, i.e., $I(x, y) = I'(x', y')$ where $(x, y) = f(x', y'; t; \delta)$. For instance, δ is the rotation angle of the first perturbation type ($t = 1$). Note that if the perturbation degree $\delta = 0$, then all perturbation types become the identity transformation, i.e., $(x, y) = (x', y')$.

For the first eleven perturbation types, we use a second-order polynomial transformation for $f(.)$:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \end{pmatrix} \begin{pmatrix} x' \\ y' \\ x'y' \\ x'^2 \\ y'^2 \end{pmatrix} \quad (1)$$

where $\{a_{ij}\}$ are functions of the perturbation type t and degree δ (see Appendix A for more details).

The last perturbation type modelling the stroke width is defined as follows. If $\delta > 0$, $I(x, y)$ is obtained by applying a dilation operator on $I'(x', y')$, i.e., $I(x, y)$ is set to black if any of $\{I'(x' = x+u, y' = y+v); |u| \leq 1, |v| \leq 1; u, v \in \mathcal{N}\}$ is black, and to white otherwise. If $\delta < 0$, $I(x, y)$ is obtained by applying an erosion operator on $I'(x', y')$, i.e., $I(x, y)$ is set to white if any of $\{I'(x' = x+u, y' = y+v); |u| \leq 1, |v| \leq 1; u, v \in \mathcal{N}\}$ is white , and to black otherwise. Dilation, respectively erosion, can be applied recursively to obtain patterns with thicker, respectively thinner, strokes, depending on the perturbation degree δ [8].

In order to catch the true perturbation type t and degree δ, we can try to cover the parameter space (t, δ) by sampling δ (t is already discrete) with a small sampling interval Δ, i.e., $\delta_t = k \cdot \Delta; t = 1, .., T; k = \pm 1, \pm 2 \; .. \; \pm K$. This approach is known as 'normalisation by exhaustive trial' [23]. However, it is very time consuming, since for each additional absolute value of k, we have to consider three values for each perturbation type $\{-k \cdot \Delta, 0, k \cdot \Delta\}$ and therefore $3^T - 1$ channels have to be added to the system. Eventually, we decided to limit the value of K to 1, with a different sampling interval Δ_t for each perturbation type, however. Moreover, we apply only one perturbation type in each channel, i.e., no mixed styles are used. In other words, for each perturbation type t, two perturbation degrees $\delta = \pm\Delta_t$ will be used. This means that we do not attempt to detect the true perturbation parameters but simply make the input image closer to one of its standard forms, instead. This is the reason why we termed our method *perturbation*. The penalisation factor for each perturbation type t is $w_t \in [0, 1]$, and is to be determined experimentally together with Δ_t in a validation phase (see Section 5.2).

In short, the perturbation-based recognition is characterised by the following parameters:

$$\begin{aligned} \Delta_t &: \textit{perturbation degree of type } t; t = 1, .., T; T = 12. \\ w_t &: \textit{downweighting factor of type } t; t = 1, .., T; T = 12. \end{aligned}$$

Taking into account the middle channel of Fig. 4 ($\delta = 0$), the total number of channels is $2T + 1$.

Notice that the above parametrisation is only one among many other possibilities, such as those in which mixed styles are used. Our choice was made mostly on the basis of simplicity in order to make the implementation and optimisation easy.

5 Experiments

We performed our experiments on two worldwide standard databases, namely, CEDAR and NIST [9, 26]. CEDAR was used in a pilot study (Section 5.1) whereas NIST, which is more than ten times larger, was used for large scale experiments (Section 5.2).

To test the perturbation method, it is necessary to have a "Classical Recogniser" (see Fig. 4). Each classical recogniser consists of two parts, namely, feature extraction and classification. We use two feature extraction methods, which are based on projection and contour histogramming, respectively (see Appendix B). In the pilot study, the distance-weighted k-nearest neighbor rule acts as classifier for both feature extraction methods [5]. In the large scale experiments, neural networks are used as classifiers for both feature types [13].

To reduce the computation time, a two-stage scheme is used in all experiments involving the perturbation method. That is, a system without perturbation is first applied. If the best score is higher than a fixed predefined threshold then the input pattern is assigned to the corresponding class. Otherwise, the perturbation method is used in the second stage. This rule is optimal in the sense that for a given error rate of the first stage, it minimises the rejection rate, i.e., the average number of patterns submitted to the second stage [2].

5.1 Pilot Study

We first performed a pilot study on isolated handwritten numerals from the CEDAR database consisting of 18468 training patterns (*br* directories) and 2213 testing patterns (*goodbs* directories) [9]. These data were collected from live mail in the U.S. and were thus totally unconstrained. In the following we will briefly present the main results of this study. These results are also reported in [7].

Three "classical" systems were implemented. The first "classical" system - called C1 in the following - uses projection-based features whereas the second - called C2 below - extracts features from contours. Both C1 and C2 use the distance-weighted k-Nearest Neighbor rule as classifier. We also implemented a third "classical" recogniser - called C3 in the following - by combining C1 and C2 [21, 3]. The combination is based on score summation, i.e., the combiner adds up the scores of each class from both systems. The recognition rates of these three systems at zero rejection level (forced choice) are presented in Table 1.

In order to test the proposed perturbation-based method, we built two systems, which will be called P1 and P2, respectively. C1 – which is the weakest method among C1, C2, and C3 – was adopted as "classical recogniser" in P1, while C3 –

Classical Systems	
C1 (Projection)	97.69
C2 (Contour)	98.19
C3 (Projection & Contour)	98.51
Perturbation-based Systems	
P1 (Projection)	98.59
P2 (Projection & Contour)	99.09
Published Systems	
S1 [Lee 93]	98.87
S2 [Lee 93]	98.46
S3 [Lee 93]	98.33
S4 [Lee 93]	97.92
S5 [Knerr 93]	97.6
S6 [Revow 93]	97.5

Table 1: Recognition rate on *goodbs* data of the CEDAR database. S1, S2, S3, and S4 are from Ref. [14], S5 from Ref. [12], and S6 from Ref. [17].

the strongest method among C1, C2, and C3 - was used in P2. In both P1 and P2, a weighted voting scheme was used as "combiner" according to Fig. 4. Recognition rates for both systems P1 and P2 at zero rejection level are reported in Table 1.

It can be seen that both P1 and P2 improved the recognition rate of the classical systems C1 and C3, respectively, and P2 outperformed all published systems on CEDAR database. However, recall that the test set is of small size (2213 samples). We therefore decided to continue our experiments with much larger databases from NIST.

5.2 Large Scale Experiments

Two databases, namely, SD3 and SD7, were provided by the American National Institute of Standards and Technology (NIST) in 1992 as parts of a conference to assess the state-of-the-art in isolated handwritten character recognition [26]. Twenty-nine groups from Europe and North America participated to compare the performance of their OCR systems. In total, 47 systems, both commercial and research, were presented. The databases contain isolated numerals (digits) as well as upper- and lower-case letters. In this report, we describe only experiments involving isolated numerals from SD3 and SD7. Most systems used SD3 for training and SD7 for testing at the conference, although a few of them were trained using proprietary databases.

Table 2 shows our partition of the databases into training, validation, and test sets. The training and validation sets as well as the first test set (Test1) are subsets of SD3. The second test set (Test2) is identical to SD7 and can thus be used for comparison with the results of the conference. The training set was used for training the neural networks and the validation set to control the stopping of the training process as well as for optimising the parameters of the perturbation method. Test1 and Test2 were run with parameters optimised on the validation set.

Similar to the pilot study, we implemented three "classical systems" CS1, CS2, and CS3. The first made use of projection-based features and a fully connected feed-forward three-layer perceptron with architecture 49:60:10 (60 hidden nodes). The second system utilised contour-based features and also a fully connected feed-forward three-layer perceptron with architecture 104:60:10 (60 hidden nodes). The third was the combination of the first two via score summation. After training, all three systems were tested on Test1 and Test2. The results are reported in Table 3, which shows that the third system performs best among the three "classical systems" for both Test1 and Test2, as can be expected from the pilot study.

The perturbation system PS – using CS3 as "classical recogniser" – was first built with score summation as "combiner", according to Fig. 4. PS parameters $(\{\Delta_t, w_t\}, t = 1, .., T = 12)$ were then obtained by minimising the empirical error rate of the validation set, using a standard optimisation technique [16]. Finally, the optimised PS was tested on Test1 and Test2. The recognition rates at zero rejection level are presented in Table 3.

For comparison, Table 4 shows the list of the top-ten systems submitted to the conference in 1992. Only those systems that were trained on the same database (SD3) are included in this list. All systems were tested on SD7 (our Test2). A more recently published system, trained and tested on the same databases, took the 8^{th} rank in the list [20], which shows that it remains a difficult task to outperform the listed systems (often from companies with decades of experience.) However, it is important to stress that our goal is not to compete with the participants of the conference, but to see whether the perturbation method can improve the recognition rate of systems that are equivalent to the state-of-the-art systems. The relatively low recognition rate of all listed systems (less than 97%) was due to the fact that the training data (SD3) and the testing data (SD7) were collected from two different populations of writers, resulting in different statistical distributions [26]. Our experiments also confirm this difference: the recognition rate of the perturbation system PS on Test1 (part of SD3) is much better than on Test2 (SD7), even though the latter is better (more than 97%) than all listed systems in Table 4.

Fig. 7 shows some numerals in Test2 (SD7) that were misrecognised by the classical system CS3, but correctly classified by the perturbation system PS.

It can be expected that system PS could further be improved by simply increasing the size of our training data. For instance, it has been found that the error rate is cut by more than half for every tenfold increase in the size of the training set from 10 to 100000 samples [20].

Database	NIST-SD3			NIST-SD7
Partition	1-40000	40001-50000	50001-223124	1-58646
Size	40000	10000	173124	58646
Use	Training	Validation	Test1	Test2

Table 2: Partition of NIST-SD3 and NIST-SD7 databases.

System	Method	Training (40000)	Validation (10000)	Test1 (173124)	Test2 (58646)	Recognition Speed
CS1	Projection	99.55%	98.54%	99.06%	95.62%	100 digits/second
CS2	Contour	99.90%	98.75%	99.34%	96.38%	25 digits/second
CS3	Projection+Contour	-----	99.02%	99.45%	96.80%	20 digits/second
PS	Perturbation	-----	99.32%	99.54%	97.10%	6 digits/second

Table 3: Recognition rates at zero rejection level and speeds for various systems on NIST databases. Training, Validation, Test1, and Test2 are subsets of NIST databases according to the partition of Table 2. The average recognition speeds were measured on a Sun SparcStation 5 (110 MHz). Notice that two fields of the table are empty because only the generalisation power (i.e. ability to classify new patterns) of the algorithm is of importance.

Rank	1	2	3	4	5	6	7	8	9	10
Recognition Rate	96.84%	96.65%	96.62%	96.57%	96.51%	96.33%	96.15%	96.12%	96.08%	95.90%

Table 4: Top-ten systems tested on SD7 (our Test2) at the 1992 NIST conference.

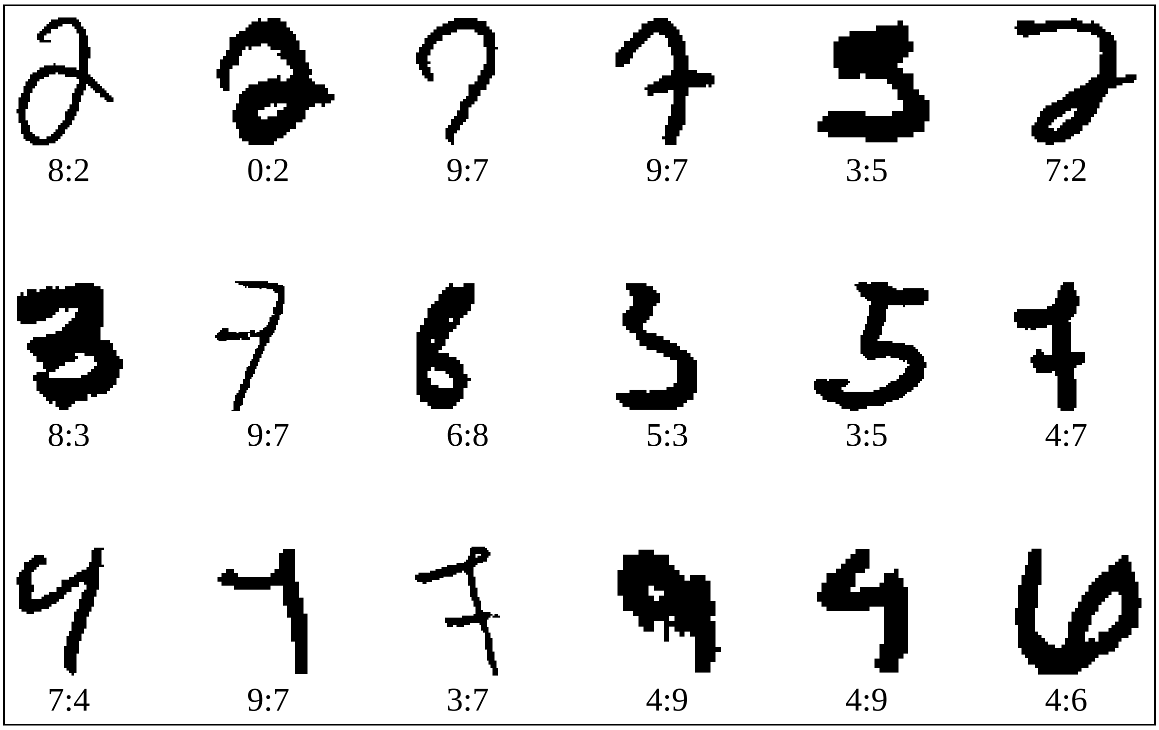

Figure 7: Samples from SD7 (our Test2) that are misrecognised by the classical system CS3, but correctly classified by the perturbation system PS. x:y means that x is the result of CS3 and y is that of PS (the true class).

6 Discussions

The results in Tables 1 and 3 show that the perturbation method consistently improves the recognition rates of "classical" recognisers, be it based on k-Nearest-Neighbor or neural network classifiers, and by using different feature types. Therefore, we believe that the perturbation method can be adapted to improve virtually any "classical" recogniser.

One drawback of the perturbation approach is its computational burden. However, this can be reduced by a multi-stage scheme. Moreover, the structure of the system is highly parallel (see Fig. 4) and therefore is perfectly suited for hardware implementation.

The main contributions of the perturbation method are twofold. First, *Observation 1* shows that *a priori* normalisation cannot always be properly achieved. In difficult cases, recognition may be necessary *prior* to normalisation. Of course, for an unknown input image, we are led to a dilemma: Should we perform normalisation or recognition first? The perturbation method provides one way to deal with the dilemma. A further step, which is out of the scope of this report, would be to use the outputs of all channels to determine the probably most correct normalisation. Second, we are aware that an attempt to *exactly* bring the input image back to one of its standard forms is an extremely difficult, if not impossible task. Therefore, the goal of the perturbation method is limited to make the input image *closer* to one of its standard forms. It turned out that this limited goal was sufficient to improve the recognition rate in the classification task. In the following, we will discuss the relations between the perturbation method and other approaches.

In a broad sense, the perturbation approach can be considered as a kind of 'generalised correlation'. The basic idea consists in detecting the maximum of the cross-correlation function between two functions, the first of which represents a training pattern and the second a testing one (perturbation is limited to translation) [24]. In the field of OCR, the earliest (to our knowledge) work that bears many resemblances to our method was described in a textbook in 1973 [23], where the author mentioned a commercial system that successively tried different binarisation thresholding until the output confidence becomes higher than a given confidence level. Such a technique is conceptually equivalent to searching for an optimum confidence in the space of threshold parameters. Compared with our approach, the main difference is that there is *no search in the perturbation method* making implementation and optimisation easier. This particular feature is also making our approach distinct from elastic matching, where usually some kind of search is required to find the minimum cost between an input sample and each prototype [25]. Another family of methods that shares some similarities with ours consists in generating additional artificial data for training by using distortion models [1, 4, 11, 28]. In these methods, the distortions parameters must be chosen very carefully so as to not include 'strange' patterns into the training set.

7 Conclusions

We have presented a new approach to handwriting recognition. First, we observed that the correct normalisation of eccentric handwriting cannot always be achieved by using the input image alone, but *a priori* information about the alphabet must be taken into account. Furthermore, we noticed that most distortions in real difficult cases are due to writing habits (styles) and instruments. Therefore, we proposed a new approach, called perturbation method, to cope with the difficult eccentric handwriting. The new approach constitutes a shift from the usual pattern recognition paradigm where normalisation is the first step prior to feature extraction and classification. Normalisation is replaced by a set of perturbation processes modelling writing habits and instruments. As a result, the subsequent operations are repeated for each perturbation type yielding a set of results that are eventually combined.

Computer simulation on handwritten numerals from two worldwide standard databases of CEDAR and NIST showed that the perturbation method consistently improves the recognition rates of state-of-the-art recognition methods. Finally, we discussed the relationships of the perturbation method to others, such as elastic matching and database enrichment.

Acknowledgments: This work is partly supported by the Union Bank of Switzerland Information Technology Laboratory (UBILAB) and the Swiss National Science Foundation. The authors wish to thank Prof. R. Marti, Mr. J. Clarinval, and Prof. H.-P. Frei of the UBILAB for initiating and continuing the collaboration. Special thanks go to our former and present students, namely, D. Niggeler, A. Grossmann, P. Steiner, M. Zimmermann, A. Tschäppeler, A. Münger, T. Studer, and D. Grana, for their helpful contributions.

A Geometric Perturbation Models

In this appendix we present the parametrisation of the eleven geometric perturbation models, which are expressed by a second-order polynomial transformation.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \end{pmatrix} \begin{pmatrix} x' \\ y' \\ x'y' \\ x'^2 \\ y'^2 \end{pmatrix} = A \cdot \begin{pmatrix} x' \\ y' \\ x'y' \\ x'^2 \\ y'^2 \end{pmatrix} \tag{2}$$

Apart from rotation, which is well-known in standard mathematics, the other perturbation models are derived using parameter identification. For example, horizontal slant is an affine transformation and therefore the nonlinear coefficients are zero, see Eq. (4). The remaining four coefficients are obtained by using the coordinates of the four corners of a square and its distorted version. This gives eight equations among which only four are sufficient to identify the four unknown coefficients. For nonlinear transformations, such as perspective views, six points lying on the boundary of the square must be used.

A.1 t=1: rotation

$$A = \begin{pmatrix} cos\delta & -sin\delta & 0 & 0 & 0 \\ sin\delta & cos\delta & 0 & 0 & 0 \end{pmatrix} \tag{3}$$

A.2 t=2: horizontal slant

$$A = \begin{pmatrix} 1 & \delta & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{4}$$

A.3 t=3: vertical slant

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \delta & 1 & 0 & 0 & 0 \end{pmatrix} \tag{5}$$

A.4 t=4: horizontal perspective

$$A = \begin{pmatrix} 1 & 0 & 0 & \delta & 0 \\ 0 & 1 & \delta & 0 & 0 \end{pmatrix} \tag{6}$$

A.5 t=5: vertical perspective

$$A = \begin{pmatrix} 1 & 0 & \delta & 0 & 0 \\ 0 & 1 & 0 & 0 & \delta \end{pmatrix} \tag{7}$$

A.6 t=6: first diagonal perspective

$$A = \begin{pmatrix} 1 & 0 & 0 & \delta & 0 \\ 0 & 1 & 0 & 0 & \delta \end{pmatrix} \tag{8}$$

A.7 t=7: second diagonal perspective

$$A = \begin{pmatrix} 1 & 0 & 0 & \delta & 0 \\ 0 & 1 & 0 & 0 & -\delta \end{pmatrix} \tag{9}$$

A.8 t=8: horizontal shrink

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & \delta & 0 & 0 \end{pmatrix} \tag{10}$$

A.9 t=9: vertical shrink

$$A = \begin{pmatrix} 1 & 0 & \delta & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{11}$$

A.10 t=10: first diagonal shrink

$$A = \begin{pmatrix} 1 & 0 & 0 & \delta & -\delta \\ 0 & 1 & 0 & -\delta & \delta \end{pmatrix} \tag{12}$$

A.11 t=11: second diagonal shrink

$$A = \begin{pmatrix} 1 & 0 & 0 & \delta & -\delta \\ 0 & 1 & 0 & \delta & -\delta \end{pmatrix} \tag{13}$$

B Feature Extraction

B.1 Projection-Based Features

The input image is normalised to a 32×32 binary image. Black pixels are projected in 4 main directions resulting in 4 histograms. Similarly, we count the number of black-to-white transition pixels in 4 main directions resulting in another 4 histograms. In addition, 8 contour profiles are computed from 8 main directions (a contour profile value is defined as the number of white pixels separating the border and the 1^{st} black pixel seen from a given direction). Thus, we obtain 16 one-dimensional functions from each of which 3 features are extracted by sub-sampling with overlapping cosinusoidal windows. The last component of the feature vector is the aspect ratio of the input image. Thus, a 49-dimensional feature vector $(16 \times 3 + 1)$ is obtained.

B.2 Contour-Based Features

The input image is normalised to a 72×54 binary image. Both inner and outer contours of the numeral are extracted. At each contour point the direction is estimated and quantised into 8 uniform quantisation intervals. The normalised space is then subdivided into $4 \times 3 = 12$ overlapping regions. In each region and for each of the 8 discrete directions, the total number of contour pixels is counted. The counting is weighted according to its position with respect to the center of the corresponding region using a separable 2-dimensional cosinusoidal window. In addition, the global 8-directional contour histogram is computed. This process yields a 104-dimensional feature vector $(8(12+1))$.

References

[1] H.S. Baird, "Document Image Defect Models," in H.S. Baird, H. Bunke, K. Yamamoto (Eds.):*Structured Document Image Analysis,* Springer Verlag, 1992, pp. 546-556.

[2] C.K. Chow, "An Optimum Character Recognition System Using Decision Functions," *IRE Transactions on Electronic Computers,* Vol. EC-6, No. 4, pp. 247-254, December 1957.

[3] B.V. Dasarathy (Editor), *Decision Fusion,* IEEE Computer Society Press, 1994.

[4] H. Drucker, R. Schapire, and P. Simard, "Boosting Performance in Neural Networks," *International Journal of Pattern Recognition and Artificial Intelligence,* Vol. 7, No. 4, pp. 705-719, 1993.

[5] S.A. Dudani, "The Distance-Weighted k-Nearest-Neighbor Rule," *IEEE Transaction on Systems, Man, and Cybernetics,* Vol. 6, No. 4, pp. 325-327, April 1976.

[6] K. Fukunaga and D.M. Hummels, "Bayes Error Estimation Using Parzen and k-NN Procedures," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. 9, No. 5, pp. 634-643, Sept. 1987.

[7] Thien M. Ha and H. Bunke, "Handwritten Numeral Recognition by Perturbation Method", *Proc. of the Fourth International Workshop on Frontiers of Handwriting Recognition,* Taipei, Taiwan, Dec. 7-9, 1994, pp. 97-106.

[8] R.M. Haralick, S.R.Sternberg, and X. Zhuang, "Image Analysis Using Mathematical Morphology," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. PAMI-9, No. 4, pp. 532-550, July 1987.

[9] J.J. Hull, "A Database for Handwritten Text Recognition Research," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. 16, No. 5, pp. 550-554, May 1994.

[10] A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall, 1989.

[11] N. Kita, "Making a Personal Recognition Dictionary from Characters Automatically Generated by Using Handwriting Model," *Proceedings of the Second International Conference on Document Analysis and Recognition,* Tsukuba, Japan, Oct. 20-22, 1993, pp. 76-81.

[12] S. Knerr and S. Sperduti, "Rejection Driven Hierarchy of Neural Network Classifiers," *Proceedings of 1993 International Symposium on Nonlinear Theory and its Applications,* Vol. 3, Hawaii, 1993.

[13] C.G.Y. Lau (Editor), *Neural Networks: Theoretical Foundations and Analysis,* IEEE Press, 1992.

[14] D.S. Lee, S.N. Srihari, "Handprinted Digit Recognition: A Comparison of Algorithms," *Pre-Proceedings of the third International Workshop on Frontiers in Handwriting Recognition,* Buffalo, New York, USA, May 25-27, 1993, pp. 153-162.

[15] G. Nagy and N. Tuong, "Normalisation Techniques for Hand-Printed Numerals," *Communications of the Association for Computing Machinery,* Vol. 13, No. 8, pp. 475-481, Aug. 1970.

[16] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical Recipes in C, The Art of Scientific Computing,* Cambridge University Press, 1988.

[17] M. Revow, C.K.I. Williams and G.E. Hinton, "Using Mixtures of Deformable Models to Capture Variations in Hand Printed Digits," *Pre-Proceedings of the third International Workshop on Frontiers in Handwriting Recognition,* Buffalo, New York, USA, May 25-27, 1993, pp. 142-152.

[18] M. Sabourin, A. Mitiche, D. Thomas, and G. Nagy, "Classifier Combination for Hand-Printed Digit Recognition," *Proceedings of the Second International Conference on Document Analysis and Recognition,* Tsukuba, Japan, Oct. 20-22, 1993, pp. 163-166.

[19] J. Schürmann, N. Bartneck, T. Bayer, J. Franke, E. Mandler, and M. Oberländer, "Document Analysis - From Pixels to Contents," *Proceedings of the IEEE,* Vol. 80, No. 7, pp. 1101-1119, 1992.

[20] S.J. Smith, M.O. Bourgoin, K. Sims, and H.L. Voorhees, "Handwritten Character Classification Using Nearest Neighbor in Large Databases," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. 16, No. 9, pp. 915-919, September 1994.

[21] C.Y. Suen, C. Nadal, R. Legault, T.A. Mai, and L. Lam, "Computer Recognition of Unconstrained Handwritten Numerals," *Proceedings of the IEEE,* Vol. 80, No. 7, pp. 1162-1180, 1992.

[22] Y.Y. Tang and C.Y. Suen, "Nonlinear Shape Restoration by Transformation Models," *Proceedings of the 10th International Conference on Pattern Recognition,* Atlantic City, New Jersey, USA, 16-21 June 1990, pp. 14-19.

[23] J.R. Ullmann, *Pattern Recognition Techniques,* London Butterworths, 1973.

[24] H.L. Van Trees, *Detection, Estimation, and Modulation Theory,* John Wiley and Sons, 1968.

[25] T. Wakahara, "Shape Matching Using LAT and its Application to Handwritten Numeral Recognition," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. 16, No. 6, pp. 618-629, June 1994.

[26] R.A. Wilkinson, J. Geist, S. Janet, P.J. Grother, C.J.C. Burges, R. Creecy, B. Hammond, J.J. Hull, N.W. Larsen, T.P. Vogl, and C.L. Wilson, *The First Census Optical Character Recognition Systems Conference,* The U.S. Bureau of Census and the National Institute of Standards and Technology, Technical Report #NISTIR 4912, Gaithersburg, MD, Aug. 1992.

[27] H. Yamada, K. Yamamoto and T. Saito, "A Nonlinear Normalization for Hand-printed Kanji Character Recognition - Line Density Equalization," *Pattern Recognition,* Vol. 23, No. 9, pp. 1023-1029, 1990.

[28] M. Yasuda, K. Yamamoto and H. Yamada, "Effect of Perturbated Correlation Method for Optical Character Recognition," *Proceedings of the Second International Conference on Document Analysis and Recognition,* Tsukuba, Japan, Oct. 20-22, 1993, pp. 830-833.